CENTRAL ASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES

https://cajmtcs.centralasianstudies.org

Volume: 03 Issue: 12 | Dec 2022 **ISSN: 2660-5309**

TECHNOLOGIES FOR IDENTIFYING VEHICLES STANDING AT TRAFFIC LIGHTS BASED ON VIDEO DATA

F.M.Nazarov
PhD, SamSU, fayzullo-samsu@mail.ru

B.Sh.Eshtemirov
Researcher intern, SamSU, eshtemirov23@gmail.com

Sh.Sh.Yarmatov
assistant, SamSU, sherzod2601@gmail.com

Abstract

At present, the traffic control frameworks in India, need insight and go about as an open-loop control framework, with no input or detecting system. Present technologies use Inductive loops and sensors to detect the number of vehicles passing by. It is a very inefficient and expensive way to make traffic lights adaptive. Using a simple CCTV camera can improve the conditions. The visual tracking of objects is amongst the most critical areas of computer vision and deep learning. The objective of this work was to develop the traffic control framework by presenting a detecting system, which gives an input to the current system, with the goal that it can adjust the changing traffic density patterns and provides a vital sign to the controller in a continuous activity. Using this method, improvement of the traffic signal switching expands the street limit, saves time for voyaging, and prevents traffic congestion. The framework additionally goes for consolidating exceptional arrangements for clearing the path for emergency vehicles. In this paper, we will detect and track vehicles on a video stream and count those going through a defined line and to ultimately give an idea of what the real-time on-street situation is across the road network. Our real target is to advance the deferral in the travel of vehicles in odd hours of the day. It uses the YOLO ("You Only Look Once") object detection technique to detect objects on each of the video frames And SORT (Simple Online and Real-time Tracking algorithm) to track those objects over different frames. Once the objects are detected and tracked over different frames, a simple mathematical calculation is applied to count the intersections between the vehicle's previous and current frame positions with a defined line.

However, accuracy drops when vehicles are either close together or have large shadows, dark vehicles do not always meet the detection criteria, and night scenes are challenging to resolve as headlight beams can create large areas that meet threshold criteria. We are focusing on the Indian roads where the traffic conditions are harsh and abrupt. We have already tested it on a live feed from various traffic prone roads and have found satisfactory results. Other Adaptive Traffic Light Systems were not able to work on the Indian traffic conditions. Our model has the edge over the other by performing at par on Indian roads. We'll increase or decrease the timer according to the conditions of the roads. This will tremendously improve the traffic conditions at a very low cost. Inductive loops are a feasible but expensive method. So, this system reduces costs and provides quick results.

A R T I C L E I N F O

Article history:
Received 6 Oct 2022
Revised form 5 Nov 2022
Accepted 10 Dec 2022

Key words: *KOMPAS 3D, congestion, Computer Vision, Deep learning, Inductive loops, YOLO, SORT, congestion, congestion density, reduction of traffic at traffic lights, video and image processing, real-time video data processing.*

---***---

1. Introduction

With the continuous development of modern society and the economy, the global consumption level continues to rise. People's demand for poultry meat, eggs, and other poultry-related products is increasing, and the livestock and poultry farming industry bear a wide scope for development. Such a large-scale demand for livestock and poultry products will inevitably lead to a continuous expansion in the scale of the farming industry. However, in the context of tight feed grain supplies, soil resources needed for breeding, and scarce water resources, the farming industry needs to continuously improve the quality and efficiency of production. Inefficient farming methods will increasingly worsen farming pollution, leading to an increased environmental burden and deviating from the concept of environmental protection. Sparrow ducks, commonly known as "hemp ducks", are the main species of domestic ducks and one of the most abundant, widely distributed, and diverse species of domestic ducks in the world. Occupying about 70% or more of the total waterfowl breeding, duck breeding is roughly divided into three types: meat, egg, and both meat and egg, which have high economic value. Large-scale hemp duck farming can meet the huge market demand for poultry meat and eggs, but at the same time, it faces pressure and challenges in many aspects [1]. As countries around the world pay attention to the ecological environment, the development of waterfowl farming has been subject to certain restrictions and regulations. Many areas have been prohibited and restricted, and the spatial range suitable for farming hemp ducks continues to shrink [2]. At present, farming is developing in the direction of intensification and ecology. Large-scale farming and higher rearing density will have a greater impact on the temperature, humidity, ventilation, harmful gases, dust, and microbial content of poultry houses. It indirectly has a series of adverse effects on the intake, growth performance, and animal welfare of birds. For example, unreasonable duck flock density rates will lead to poor living conditions, causing physiological diseases, such as body abrasions, skin damage, and fractures. Considering animal behavior, such as pecking and fighting among a species, the unreasonable density will bear a negative impact on the efficiency and economy of the livestock and poultry industry [3,4]. From the above information, it can be concluded that rearing density is one of the key factors affecting livestock and poultry production on a large scale, as well as animal welfare, and the key to solving the problem of improving breeding efficiency lies in the real-time monitoring of breeding density and the reasonable scheduling of the spatial quantity of the flock: our work is focused on the former. At present, in the hemp duck farming industry, much of the counting is carried out manually or by artificial machinery, which are both very laborious. When hemp duck flocks are in motion, it further increases the difficulty of manual counting, thus affecting the breeding efficiency. In essence, the density of the hemp duck flock depends on the size of the effective activity space as well as the population size, and given the constant limitations of the current breeding area, the main factor affecting the problem is therefore the number of hemp ducks. For this reason, we focused on the hemp duck flock count. With the development of technology, monitoring equipment plays a huge role in farming. There are various methods to monitor the behavior of individual animals, such as the insertion of chips to record physiological data, the use of wearable sensors, and (thermal) imaging techniques. Some methods employ wearable sensors attached to the feet of birds to measure their activity, but this may have an additional impact on the monitored animals [5–7]. In particular, in commercial settings, technical limitations and high costs lead to the low feasibility of such methods. Therefore, video assessment based on optical flow would be an ideal method to monitor poultry behavior and physiology [4]. Initially, many surveillance videos were manually observed, inefficient, and relied on the staff's empirical judgment without standards [8]. However, in recent years, due to the advent of the era of big data and the rapid development of computer graphics cards, the computing power of computers has been increasing, accelerating the development of artificial intelligence. Research related to artificial intelligence is increasing, and computer vision is becoming more and more widely used in animal detection. For example, the R-CNN proposed by Girshick et al., in 2014 introduced a two-stage detection method for the first time. This method uses deep convolutional networks to obtain excellent target detection accuracy, but its many redundant operations greatly increase space and time costs, and is difficult to deploy in actual duck farms [9,10]. Law et al., proposed a single-stage detection method, CornerNet, and a new pooling method: corner pool. However, the method, based on key points, often encounters a large number of incorrect object bounding boxes, which limits its performance and cannot meet the high performance requirements of the duck breeding model [11]. Duan et al., constructed the CenterNet framework on the basis of CornerNet to improve the accuracy and recall and designed two custom modules with stronger robustness to feature-level noise, but the anchor-free method is a process with key point combinations of the first two, and due to the simple network structure, time-consuming processing, low rate, and unstable measurement results, it cannot meet the requirements of high performance and high accuracy rate needed in the industrial farming of hemp ducks [12].

Our work uses a single-stage object detection algorithm, which only needs to extract features once to achieve object detection, and its performance is higher compared to the multi-stage algorithm. At present, the mainstream single-stage target detection algorithms mainly include the YOLO series, Single Shot MultiBox Detector (SSD), RetinaNet, etc. In this paper, we transfer and apply the idea of crowd counting based on CNN to the problem of counting ducks [13,14]. Along with the output of the detection results, we embedded an object counting module to respond to industrialization needs. Object counting is also a common task in the computer vision community. Object counting can be divided into multi-category object counting and single-category object counting; this work employed single-category counting of a flock of hemp ducks [15–18]. The objectives that this paper hopes to achieve are: (1) We built a new large-scale dataset of drake images and named it the "Hemp Duck Dataset". The Hemp Duck Dataset contains 1500 labels for the whole body frame and head frame for duck target detection. The Hemp Duck Dataset was released for the first time by the team. We have made it public and provide the access method at the end of the article. (2) This study constructed a comprehensive working baseline, including hemp duck identification, hemp duck object detection, and hemp duck image counting, to realize the intelligent breeding of hemp ducks. (3) This project model introduced the CBAM module to build the CBAM-YOLOv7 algorithm. 2. Materials and Methods 2.1. Acquisition of Materials The hemp duck is one of the most abundant, widely distributed, and diverse species of domestic ducks in China, with the characteristics of small size, feed saving, and high egg production efficiency, which is of great research value. We used the DJI Pocket 2, an extremely adaptable and flexible miniature gimbal camera, to capture the image and video datasets used in this study. Data were collected from the original waterfowl farm in Ya'an, Sichuan Province, China, founded by Professor Lin-Quan Wang, a renowned waterfowl breeder from Sichuan Agricultural University. In the process of preparing the dataset, we first collected data from 10 different hemp duck houses by changing the image shooting angle and distance several times. Then, we manually screened and discarded some data with high repetition and some redundant data that were not captured due to the obstruction of the hemp ducks' house. In the end, our dataset contained a total of 1500 images, including 1300 images in the training set and 200 images in the test set. Figure 1 shows the analysis of the challenges posed by non-maximum suppression for the hemp duck detection, identification, and counting tasks. Figure 2 shows an example of a dataset labeling effort. In the prediction phase of the object detection work, the network output multiple candidate anchor boxes, but many of them were overlapping around the same object, as shown in Figure 1b. Non-maximum suppression was able to retain the best one among this group of candidate anchor boxes, as shown in Figure 1c. We named two different ducks hemp duck A and hemp duck B. When hemp duck A and hemp duck B are too close, the prediction box of hemp duck A may be eliminated due to the screening of non-maximum intrusion. Therefore, it is a challenge to accurately estimate the number of dense Hemp Duck Datasets with inclusion. Since labeling the whole hemp duck body resulted in many overlapping labeling boxes, which affected the accuracy of individual hemp duck counting, we chose the method of labeling only the hemp duck head and conducted a comparison experiment between the two.

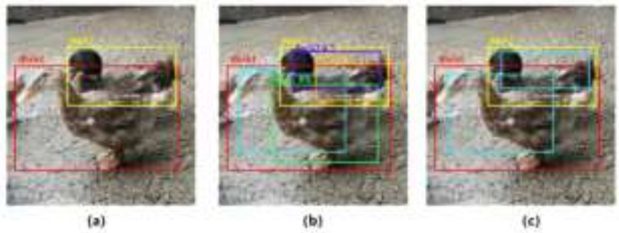

Figure 1. (a) Two ground truth boxes of hemp ducks; (b) output prediction boxes of the simulated network for the two hemp ducks; (c) effect of removing the redundant prediction boxes after nonmaximum suppression.

Figure 2. (a) Example of data annotation of the whole body of hemp ducks; (b) example of data annotation of hemp ducks with only the head annotated.

About Yolo technology

The YOLOv7 algorithm is making big waves in the computer vision and machine learning communities. The newest YOLO algorithm surpasses all previous object detection models and YOLO versions in both speed and accuracy. It requires several times cheaper hardware than other neural networks and can be trained much faster on small datasets without any pre-trained weights. Hence, YOLOv7 is expected to become the industry standard for object detection in the near future, surpassing the previous state-of-the-art for real-time applications – YOLO v4. Our Background: At viso.ai, we provide the most advanced enterprise computer vision deployment platform Viso Suite. The no-code software infrastructure is used by leading organizations to train YOLOv7 models and build computer vision applications. In this article, we will provide the basics of how YOLOv7 works and what makes it the best object detector algorithm available. The Computer Vision Platform Viso Suite supports YOLOv7 out-of-the-box to build custom applications. YOLO Real-Time Object Detection What is real-time object detection? In computer vision, real-time object detection is a very important task that is often a key component in computer vision systems. Applications that use real-time object detection models include video analytics, robotics, autonomous vehicles, multi-object tracking and object counting, medical image analysis, and so on. An object detector is an object detection algorithm that performs image recognition tasks by taking an image as input and then predicting bounding boxes and class probabilities for each object in the image (see the example image below). Most algorithms use a convolutional neural network (CNN) to extract features from the image to predict the probability of learned classes. YOLOv7 applied for computer vision in Aviation – built on Viso Suite What is YOLO in computer vision? YOLO stands for “You Only Look Once”, it is a popular family of real-time object detection algorithms. The original YOLO object detector was first released in 2016. It was created by Joseph Redmon, Ali Farhadi, and Santosh Divvala. At release, this architecture was much faster than other object detectors and became state-of-the-art for real-time computer vision applications. Since then, different versions and variants of YOLO have been proposed, each providing a significant increase in performance and efficiency. The versions from YOLOv1 to the popular YOLOv3 were created by then-graduate student Joseph Redmon and advisor Ali Farhadi. YOLOv4 was introduced by Alexey Bochkovskiy, who continued the legacy since Redmon had stopped his computer vision research due to ethical concerns. YOLOv7 is the latest official YOLO version created by the original authors of the YOLO architecture. We expect that many commercial networks will move directly from YOLOv4 to v7, bypassing all the other numbers. Unofficial YOLO versions There were some controversies in the computer vision community whenever other researchers and companies published their models as YOLO versions. A popular example is YOLOv5 which was created by the company Ultralytics. It’s similar to YOLOv4 but uses a different framework, PyTorch, instead of DarkNet. However, the creator of YOLOv4, Alexey Bochkovskiy provided benchmarks comparing YOLOv4 vs. YOLOv5, showing that v4 is equal or better. Another example is YOLOv6 which was published by the Chinese company Meituan (hence the MT prefix of YOLOv6). And there is also an unofficial YOLOv7 version that was released in the year before the official YOLOv7 (there are two YOLOv7’s). Both YOLOv5 and YOLOv6 are not considered part of the official YOLO series but were heavily inspired by the original one-stage YOLO architecture. Critics argue that companies try to benefit from the YOLO hype and that the papers were not adequately peer-reviewed or tested under the same conditions. Hence, some say that the official YOLOv7 should be the real YOLOv5. YOLOv7 object detection in a dense scene – Viso Suite Real-time object detectors and YOLO versions Currently, state-of-the-art real-time object detectors are mainly based on YOLO and FCOS (Fully Convolutional One-Stage Object Detection). The best performing object detectors are: YOLOv3 model, introduced by Redmon et al. in 2018 YOLOv4 model, released by Bochkovskiy et al. in 2020, YOLOv4-tiny model, research published in 2021 YOLOR (You Only Learn One Representation) model, published in 2021 YOLOX model, published in 2021 NanoDet-Plus model, published in 2021 PP-YOLOE, an industrial object detector, published in 2022 YOLOv5 model v6.1 published by Ultralytics in 2022 YOLOv7, published in 2022 How to run object detection efficiently at the Edge Running object detection in real-world computer vision applications is hard. Key challenges include the allocation of computing resources, system robustness, scalability, efficiency, and latency. In addition, ML computer vision requires IoT communication (see AIoT) for data streaming with images as input and detections as output. To overcome those challenges, the concept of Edge AI has been introduced, which leverages Edge Computing with Machine Learning (Edge ML, or Edge Intelligence). Edge AI modes ML processing from the cloud closer to the data source (camera). Thus, Edge AI applications form distributed edge systems with multiple, connected edge devices or virtual edge nodes (MEC or cloud). The Advantages of Edge AI for real-time object detectors The computing device that executes object detection is usually some edge device with a CPU or GPU processor, as well as neural processing units (NPU) or vision accelerators. Such NPU devices are increasingly popular AI hardware for computer vision inferencing, for example: Neural compute stick or NCS (Intel) Jetson AI edge devices (Nvidia) Apple neural engine (Apple) Coral Edge TPU (Google) Neural processing engine (Qualcomm) More recently, the design of efficient object detection architectures has focused on models that can be used on CPU for scalable edge applications. Such models are mainly based on MobileNEt, ShuffleNet, or GhostNet. Other mainstream object detectors have been optimized for GPU computing, they commonly use ResNet, DarkNet, or DLA architectures. The end-to-end Edge AI vision platform Viso Suite lets you enroll and manage edge devices, and use any AI hardware, camera, and processor to run computer vision at the Edge. Request a demo here. What is YOLOv7 YOLOv7 is the fastest and most accurate real-time object detection model for computer vision tasks. The official YOLOv7 paper named “YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors” was released in July 2022 by Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. The YOLOv7 research paper has become immensely popular in a matter of days. The source code was released as open source under the GPL-3.0 license, a free copyleft license, and can be found in the official YOLOv7 GitHub repository that was awarded over 4.3k stars in the first month after release. There is also a complete appendix of the YOLOv7 paper. YOLOv7 used in an application of computer vision in construction – built on Viso Suite The differences between the basic YOLOv7 versions The different basic YOLOv7 models include YOLOv7, YOLOv7-tiny, and YOLOv7-W6: YOLOv7 is the basic model that is optimized for ordinary GPU computing. YOLOv7-tiny is a basic model optimized for edge GPU. The suffix “tiny” of computer vision models means that they are optimized for Edge AI and deep learning workloads, and more lightweight to run ML on mobile computing devices or distributed edge servers and devices. This model is important for distributed real-world computer vision applications. Compared to the other versions, the edge-optimized YOLOv7-tiny uses leaky ReLU as the activation function, while other models use SiLU as the activation function. YOLOv7-W6 is a basic model optimized for cloud GPU computing. Such Cloud Graphics Units (GPUs) are computer instances for running applications to handle massive AI and deep learning workloads in the cloud without requiring GPUs to be deployed on the local user device. Other variations include YOLOv7-X, YOLOv7-E6, and YOLOv7-D6, which were obtained by applying the proposed compound scaling method (see YOLOv7 architecture further below) to scale up the depth and width of the entire model. YOLOv7-mask The integration of YOLOv7 with BlendMask is used to perform instance segmentation. Therefore, the YOLOv7 object detection model was fine-tuned on the MS COCO instance segmentation dataset and trained for 30 epochs. It achieves state-of-the-art real-time instance segmentation results. Architecture for image segmentation with YOLOv7-mask – Source YOLOv7-mask, for instance segmentation tasks – Source YOLOv7-pose The integration of YOLOv7 with YOLO-Pose allows keypoint detection for Pose Estimation. The authors fine-tuned a YOLOv7-W6 people detection model on the MS COCO keypoint detection dataset and achieved state-of-the-art real-time pose estimation performance. The architecture for pose estimation with YOLOv7-pose – Source Examples of pose estimation with YOLOv7 – Source What is new with YOLOv7? YOLOv7 provides a greatly improved real-time object detection accuracy without increasing the inference costs. As previously shown in the benchmarks, when compared to other known object detectors, YOLOv7 can effectively reduce about 40% parameters and 50% computation of state-of-the-art real-time object detections, and achieve faster inference speed and higher detection accuracy. In general, YOLOv7 provides a faster and stronger network architecture that provides a more effective feature integration method, more accurate object detection performance, a more robust loss function, and an increased label assignment and model training efficiency. As a result, YOLOv7 requires several times cheaper computing hardware than other deep learning models. It can be trained much faster on small datasets without any pre-trained weights. YOLOv7 video analytics application of computer vision in Smart City – Built with Viso Suite The authors train YOLOv7 using the MS COCO dataset without using any other image datasets or pre-trained model weights. Similar to Scaled YOLOv4, YOLOv7 backbones do not use Image Net pre-trained backbones (such as YOLOv3). The YOLOv7 paper introduces the following major changes. Later in this article, we will describe those architectural changes and how YOLOv7 works. YOLOv7 Architecture Extended Efficient Layer Aggregation Network (E-ELAN) Model Scaling for Concatenation based Models Trainable Bag of Freebies Planned re-parameterized convolution Coarse for auxiliary and fine for lead loss What are Freebies in YOLOv7? Bat-of-freebies features (more optimal network structure, loss function, etc.) increase accuracy without decreasing detection speed. That’s why YOLOv7 increases both speed and accuracy compared to previous YOLO versions. The term was introduced in the YOLOv4 paper. Usually, a conventional object detector is trained off-line. Consequently, researchers always like to take this advantage and develop better training methods that can make the object detector receive better accuracy without increasing the inference cost (read about computer vision costs). The authors call these methods that only change the training strategy or only increase the training cost a “bag of freebies”. Where can I quickly test YOLOv7? Here is a very fast way to test the new YOLOv7 deep learning model directly on Hugging Face: Find it here. This allows you to (1) upload your own images from your local device, (2) select a YOLOv7 model, and (3) generate an output image with label boxes. Since the DL model was trained on the COCO dataset, it will perform image recognition to detect the default COCO classes (find them in our guide about MS COCO). Test with a free YOLOv7 demo on Hugging Face Performance of YOLOv7 Object Detection The YOLOv7 performance was evaluated based on previous YOLO versions (YOLOv4 and YOLOv5) and YOLOR as baselines. The models were trained with the same settings. The new YOLOv7 shows the best speed-to-accuracy balance compared to state-of-the-art object detectors. In general, YOLOv7 surpasses all previous object detectors in terms of both speed and accuracy, ranging from 5 FPS to as much as 160 FPS.

The YOLO v7 algorithm achieves the highest accuracy among all other real-time object detection models – while achieving 30 FPS or higher using a GPU V100. Comparison with other real-time object detectors: YOLOv7 achieves state-of-the-art (SOTA) performance. – Source Compared to the best performing Cascade-Mask R-CNN models, YOLOv7 achieves 2% higher accuracy at a dramatically increased inference speed (509% faster). This is impressive because such R-CNN versions use multi-step architectures that previously achieved significantly higher detection accuracies than single-stage detector architectures. YOLOv7 outperforms YOLOR, YOLOX, Scaled-YOLOv4, YOLOv5, DETR, ViT Adapter-B, and many more object detection algorithms in speed and accuracy. Comparison of baseline object detectors YOLOR and YOLOv4 with the new YOLOv7. – Source YOLOv7 vs YOLOv4 comparison In comparison with YOLOv4, YOLOv7 reduces the number of parameters by 75%, requires 36% less computation, and achieves 1.5% higher AP (average precision). Compared to the edge-optimized version YOLOv4-tiny, YOLOv7-tiny reduces the number of parameters by 39%, while also reducing computation by 49%, while achieving the same AP. YOLOv7 vs YOLOR comparison Compared to YOLOR, Yolov7 reduces the number of parameters by 43% parameters, requires 15% less computation, and achieves 0.4% higher AP. When comparing YOLOv7 vs. YOLOR using the input resolution 1280, YOLOv7 achieves an 8 FPS faster inference speed with an increased detection rate (+1% AP). When comparing YOLOv7 with YOLOR, the YOLOv7-D6 achieves a comparable inference speed, but a slightly higher detection performance (+0.8% AP). YOLOv7 vs YOLOv5 comparison Compared to YOLOv5-N, YOLOv7-tiny is 127 FPS faster and 10.7% more accurate on AP. The version YOLOv7-X achieves 114 FPS inference speed compared to the comparable YOLOv5-L with 99 FPS, while YOLOv7 achieves a better accuracy (higher AP by 3.9%). Compared with models of a similar scale, the YOLOv7-X achieves a 21 FPS faster inference speed than YOLOv5-X. Also, YOLOv7 reduces the number of parameters by 22% and requires 8% less computation while increasing the average precision by 2.2%. Comparing YOLOv7 vs. YOLOv5, the YOLOv7-E6 architecture requires 45% fewer parameters compared to YOLOv5-X6, and 63% less computation while achieving a 47% faster inference speed. YOLOv7 vs PP-YOLOE comparison Compared to PP-YOLOE-L, YOLOv7 achieves a frame rate of 161 FPS compared to only 78 FPS with the same AP of 51.4%. Hence, YOLOv7 achieves an 83 FPS or 106% faster inference speed. In terms of parameter usage, YOLOv7 is 41% more efficient. YOLOv7 vs YOLOv6 comparison Compared to the previously most accurate YOLOv6 model (56.8% AP), the YOLOv7 real-time model achieves a 13.7% higher AP (43.1% AP) on the COCO dataset. Any comparing the lighter Edge model versions under identical conditions (V100 GPU, batch=32) on the COCO dataset, YOLOv7-tiny is over 25% faster while achieving a slightly higher AP (+0.2% AP) than YOLOv6-n. Comparison of the best real-time object detectors from the official YOLOv7 paper. – Source Performance comparison YOLOv7 vs. YOLOR vs. YOLOX vs. YOLOv5 and Vit Transformers. – Source YOLOv7 Architecture The YOLOv7 architecture is based on previous YOLO model architectures, namely YOLOv4, Scaled YOLOv4, and YOLO-R. In the following, we will provide a high-level overview of the most important aspects that are detailed in the YOLOv7 paper. To learn more about deep learning architectures, check out our article about the three popular types of Deep Neural Networks. Extended Efficient Layer Aggregation Network (E-ELAN) The computational block in the YOLOv7 backbone is named E-ELAN, standing for Extended Efficient Layer Aggregation Network. The E-ELAN architecture of YOLOv7 enables the model to learn better by using "expand, shuffle, merge cardinality" to achieve the ability to continuously improve the learning ability of the network without destroying the original gradient path. YOLOv7 Compound Model Scaling The main purpose of model scaling is to adjust key attributes of the model to generate models that meet the needs of different application requirements. For example, model scaling can optimize the model width (number of channels), depth (number of stages), and resolution (input image size). In traditional approaches with concatenation-based architectures (for example, ResNet or PlainNet), different scaling factors cannot be analyzed independently and must be considered together. For instance, scaling-up model depth will cause a ratio change between the input channel and output channel of a transition layer, which in turn may lead to a decrease in hardware usage of the model. This is why YOLOv7 introduces compound model scaling for a concatenation-based model. The compound scaling method allows to maintain the properties that the model had at the initial design and thus maintain the optimal structure. And this is how compound model scaling works: For example, scaling the depth factor of a computational block also requires a change in the output channel of that block. Then, width factor scaling is performed with the same level of change on the transition layers. How YOLOv7 works: Compound model scaling – Source Planned re-parameterized convolution While RepConv has achieved great performance in VGG architectures, the direct application in ResNet or DenseNet leads to significant accuracy loss. In YOLOv7, the architecture of planned re-parameterized convolution uses RepConv without identity connection (RepConvN). The idea is to avoid that there is an identity connection when a convolutional layer with residual or concatenation is replaced by re-parameterized convolution. Planned re-parameterized model in YOLOv7 architecture: The RepConv of a layer with residual or concatenation connections should not have an identity connection and is thus replaced by RepConvN that has no identity connections. Coarse for auxiliary and fine for lead loss A YOLO architecture contains a backbone, a neck, and a head. The head contains the predicted model outputs. Inspired by Deep Supervision, a technique often used in training deep neural networks, YOLOv7 is not limited to one single head. The head responsible for the final output is called the lead head, and the head used to assist training in the middle layers is named auxiliary head. In addition, and to enhance the deep network training, a Label Assigner mechanism was introduced that considers network prediction results together with the ground truth and then assigns soft labels. Compared to traditional label assignment that directly refers to the ground truth to generate hard labels based on given rules, reliable soft labels use calculation and optimization methods that also consider the quality and distribution of prediction output together with the ground truth.

In this paper, a recognition and detection algorithm based on computer vision is proposed for object detection and population statistics of farm ducks. By using this algorithm, breeders can obtain the quantity and behavior dynamics of mallard ducks in real time so as to realize the rapid management and strategy formulation of farms, optimize the reproduction rate and growth of ducks, and help to maximize the economic benefits. In view of the small density of individuals in the duck population and the real-time requirement of population statistics, we chose the latest Yolov7 model. You Only Look Once (Yolov7) is a single-stage object detection algorithm. The Yolov7 model preprocessing method is integrated with Yolov5, and the use of Mosaic data augmentation is suitable for small object detection [13,14,21]. In terms of architecture, extended ELAN (E-ELAN) based on ELAN is proposed. Expand, shuffle, and merge cardinality are used to continuously enhance the learning ability of the network without destroying the original gradient path. Group convolution is used to expand the channel and cardinality of the computing block in the architecture of the computing block. Different groups of computational blocks are guided to learn more diverse features [13]. Then, it focuses on some optimization modules and methods known as trainable “bag-of-freebies” [13], including the following: 1. RepConv without identity connection is used to design the architecture of planned reparametrized convolution, which provides more gradient diversity for different feature maps [22]. 2. The auxiliary detection head is introduced, and the soft labels generated by the optimization process are used for lead head and auxiliary head learning. Therefore, the soft labels generated from it should better represent the distribution and correlation between source data and object and obtain more accurate results [23]. (1) The batch normalization layer is directly connected to the convolution layer so that the normalized mean and variance of the batch are integrated into the deviation and weight of the convolution layer in the inference stage. (2) By using the addition and multiplication method of implicit knowledge in YOLOR combined with the convolution feature map, it can be simplified into vectors by precomputation in the inference stage so as to combine with the deviation and weight of the previous or subsequent convolution layer [24]. (3) The EMA model is used purely as the final inference model. Finally, real-time object detection can greatly improve the detection accuracy without increasing the reasoning cost so that the speed and accuracy in the range of 5–160 FPS exceed all known object detectors, and fast response and accurate prediction of object detection can be achieved [25].

References

1. Akhatov A., Nazarov F. Rashidov A. Mechanisms of Information Reliability In big data and Blockchain Technologies . International conference on information science and communication technologies. November 4, 5, 6. ICISCT 2021 (IEEE), art. no . 9670052 DOI: 10.1109 / ICISCT52966.2021.9670052.
2. Akhatov A., Nazarov F. Rashidov A. Increasing data reliability by using big data parallelization mechanisms. International conference on information science and communication technologies. November 4, 5,6. ICISCT 2021 (IEEE), art. no . 9670387 DOI: 10.1109 / ICISCT52966.2021.9670387.
3. F.Nazarov, Sh.Yarmatov, B.Eshtemirov ALGORITHMS TO INCREASE DATA RELIABILITY IN VIDEO TRANSKRIBATION
4. F.Nazarov, Sh.Yarmatov, B.Eshtemirov Video data processing methodology for investigation
5. Munish Sabharwal, F.Nazarov, B.Eshtemirov Effectiveness Analysis Of Blockchain Mechanisms Using Consensus Algorithms
6. Abdoli, A.; Alaee, S.; Imani, S.; Murillo, A. Fitbit for chickens? Time series data mining can increase the productivity of poultry farms. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, CA, USA, 6–10 July 2020; pp. 3328–3336.
7. Van der Sluis, M.; de Haas, Y.; de Klerk, B.; Rodenburg, T.B.; Ellen, E.D. Assessing the activity of individual group-housed broilers throughout life using a passive radio frequency identification system—A validation study. Sensors 2020, 20, 3612. [CrossRef] [PubMed]
8. Gu, Y.; Wang, S.; Yan, Y.; Tang, S.; Zhao, S. Identification and Analysis of Emergency Behavior of Cage-Reared Laying Ducks Based on YoloV5. Agriculture 2022, 12, 485. [CrossRef]
9. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
10. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans. Pattern Anal. Mach. Intell. 2015, 37, 1904–1916. [CrossRef] [PubMed]
11. Law, H.; Deng, J. Cornernet: Detecting objects as paired keypoints. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 734–750.
12. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. Centernet: Object detection with keypoint triplets. arXiv 2019, arXiv:1904.08189.
13. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. arXiv 2022, arXiv:2207.02696.
14. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. arXiv 2020, arXiv:2004.10934.